

1. Introduction

2. Aim

3. Methodology

4. Results

5. Conclusion

6. References

7. Appendix

8. Acknowledgements

9. Bibliography

10. Glossary

11. Index

12. Appendix

13. Acknowledgements

14. Bibliography

15. Glossary

16. Index

17. Appendix

18. Acknowledgements

19. Bibliography

Expt 15:- Implement a Yolo model to detect object

Aim:-

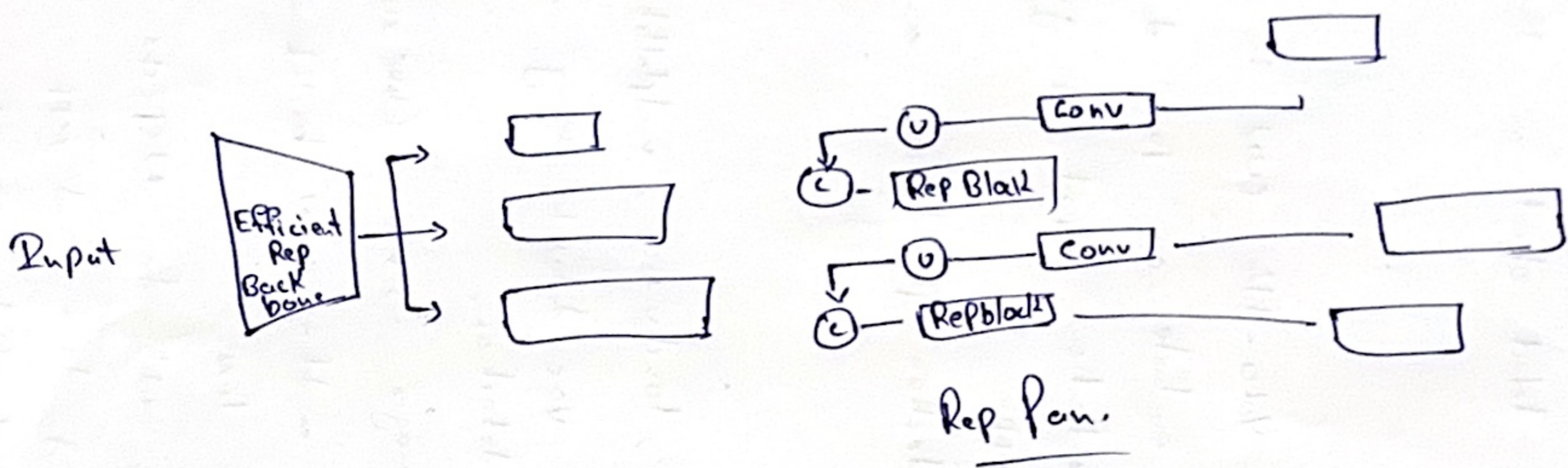
To implement a simplified Yolo-Style object model using Python to detect object on the common dataset objects.

Objective:-

- 1) Build a tiny-Yolo-like CNN in Python.
- 2) Train and evaluate on Pascal dataset.
- 3) Generate classification report and graph for loss and mAP.
- 4) Observe detection performance and analyse result.

Pseudocode:-

- 1) Import torch, torchvision, matplotlib, sklearn.
- 2) Load Pascal voc dataset using torchvision, datasets, voc detection.
- 3) PreProcess images  $\rightarrow$  resize (use xutils), normalization.
- 4) Build Yolo model  $\rightarrow$  CNN backbone.
- 5) Define Yolo loss.
- 6) Train the model for N epochs, track loss.
- 7) Plot training loss and mAP.
- 8) Print final classification report.



## YOLO Model

### Observation:-

- The training loss should gradually decrease across epochs showing that the YOLO model is learning basic localization.
- Because the dataset is tiny, detection equality will be the goal is to understand the YOLO Pipeline.
- Increasing the dataset subset and epochs will yield much better bounding box predictions.

### Result:-

The model successfully runs, loss curve showing decreasing loss values. Provided evaluation matrix and visualization.

*[Signature]*